Tuning Jordan Algebra Artificial Chemistries with Probability Spawning Functions

Penelope Faulkner[1,3], Angelika Sebald[1,3] and Susan Stepney[2,3]

[1]Department of Chemistry, University of York, UK
[2]Department of Computer Science, University of York, UK
[3]York Centre for Complex Systems Analysis
pf550@york.ac.uk

Abstract

Natural chemistry deals with non-deterministic processes, and this is reflected in some artificial chemistries. We can tune these artificial systems by manipulating the functions that define their probabilistic processes. In this work we consider different probabilistic functions for particle linking, applied to our Jordan Algebra Artificial Chemistry. We use five base functions and their variations to investigate the possible behaviours of the system, and try to connect those behaviours to different traits of the functions. We find that, while some correlations can be seen, there are unexpected behaviours that we cannot account for in our current analysis. While we can set and manipulate the probabilities in our system, it is still complex and still displays emergent behaviour that we can not fully control.

Introduction

We present an exploration of the space of and effect of 'probability spawning functions' in artificial chemistries (AChems). We use Jordan Algebra AChem (JA AChem) as an example of the use of probability spawning functions (psfs).

Natural chemistry is a probabilistic process where environmental variables (such as temperature) can affect the probability of bonding. AChems are inspired, to a greater or lesser degree, by natural chemistry, but there are many probabilistic attributes of natural chemistries that are often made deterministic in artificial chemistries, such as some aspects of movement, linking, and decomposition of links. This ignores a key feature of natural chemistry that could help our AChems to exhibit more complex behaviour.

Different AChems take different approaches in terms of determinism. Some systems, such as Hutton (2002), always link particles that encounter each other and match a linking rule. Young and Neshatian (2015) investigate different approaches by which reactants are chosen for linking, but the linking is then deterministic. There are a few AChems that have probabilistic processes, such as the selection of parameter sets in a recipe in SwarmChem (Sayama, 2009), or probabilistic movement in the 2D membrane AChem (Ono and Ikegami, 2001).

Here we focus on our Jordan Algebra Artificial Chemistry, JA-AChem (Faulkner et al., 2016), which is a *subsymbolic* AChem, one where various particle and linking properties emerge from the underlying structure of the particles (Faulkner et al., 2017). Subsymbolic AChems may include stochastic processes, such as stochastic decay in Stringmol (Hickinbotham et al., 2012; Clark et al., 2017), but others are deterministic, for example RBN world (Faulconbridge et al., 2010, 2011) and Spiky-RBN (Krastev et al., 2016, 2017).

JA-AChem's properties emerge from its underlying algebraic structures. It uses 3×3 Hermitian matrices to represent both atomic and composite particles, and the Jordan matrix product (McCrimmon, 2006) to represent linked particles:

$$A \bullet B = \tfrac{1}{2}(AB + BA) \tag{1}$$

This provides a commutative non-associative product, which allows non-trivial isomers (Faulkner et al., 2016). JA-AChem uses the matrices' eigenvalues and vectors to define the probability of a link forming, and for other parts of our system. We partition a base set of matices (Eqn 2) into 69 equivalence classes based on their eigenvalues, and take one representative atom from each class, to form our basic atomic set.

$$\left\{ A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} : a_{ij} \in \pm1, 0, \pm i, \pm1 \pm i \right\} \tag{2}$$

JA-AChem has a few probabilistically driven processes, including particle linking and decomposition. In Faulkner et al. (2016) the design of these probabilistic processes is chosen in a somewhat arbitrary manner. Here we explore how different choices for these processes may be exploited to "tune" our system in different ways and towards different behaviours, while still keeping an open mind about systemic properties.

Probability Spawning Functions

We introduce a set of functions $\mathcal{F}$ for our system that use the state of the system to yield a probability for some event or

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

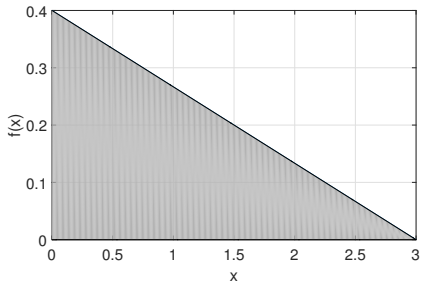

Figure 1: Plot of the function $f(x)$ where x is the absolute difference in the most similar eigenvalues of two particles. The shaded area shows the combination of x and z for which linking succeeds.

reaction. We call these *probability spawning functions* (psf), as they are used to 'spawn' specific probabilities for linking. They are not probability *distribution* functions, as there is no requirement for their integral to be 1. A real-valued psf, $f \in \mathcal{F}$, is such that $f(x) \in [0, 1]$ for all states of the system $x \in \Sigma$.

Consider a pair of eigenvalues, e_a and e_b, one from each of the reactant particles. Define $x = |e_a - e_b| \geq 0$ as the only parameter of the psf. We take the absolute value of the difference in order to preserve commutativity of the link (Faulkner et al., 2016).

For illustration purposes, consider a simple psf, the linear decreasing function with a cut-off (Figure 1):

$$f(x) = \begin{cases} \frac{2}{15}(3 - x) & \text{for} \quad x \leq 3 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The constants 3 and $2/15$ are chose to give behaviour easily comparable to that of the hard-wired psf used in the JAAChem in Faulkner et al. (2016). With this particular choice of f, the more similar the eigenvalues, the larger the probability of linking, and two particles all of whose eigenvalues differ by more than 3 have zero probability of linking.

This function f is used to determine the success or failure of a linking attempt (ignoring for now any other factors) as follows. Choose the relevant eigenvalues e_a and e_b from each particle (see later for how this choice is made). Generate a random number z in the interval [0,1] using a uniform distribution. The linking attempt succeeds if $z \leq f(x)$, that is, if (x, z) falls in the shaded area of Figure 1.

Other Probability Spawning Functions

The linear function f of equation (3) is just one possible function. There are many others we can use to determine linking probability. For example, a link would always be formed using the constant function $c(x) = 1$.

There are many aspects of such functions that we can consider. Here we focus on three: total area, position of peak, and size of tail. To help us identify effects on the system caused by each of these aspects, we start with a basic set of functions that cover a variety of different possibilities for each of these. Our basic functions are shown in Figure 2.

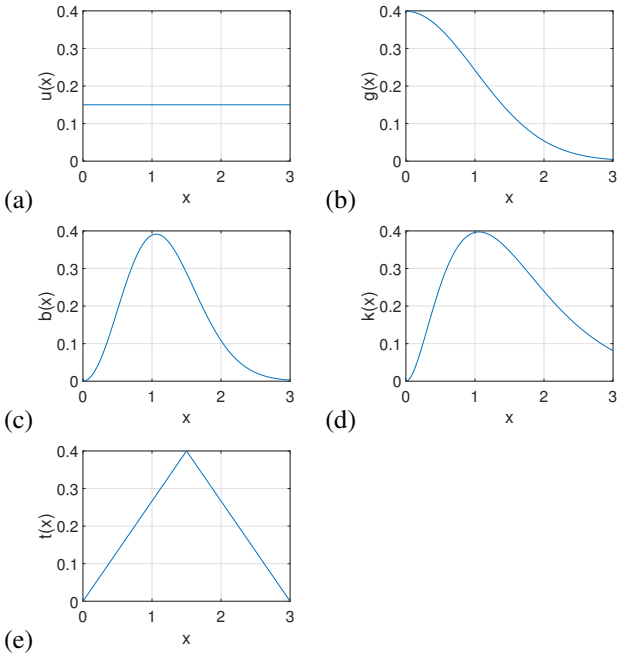

Figure 2: Plots of the five basic functions used in JA AChem. (a) uniform $u(x)$; (b) Gaussian $g(x)$; (c) Maxwell-Boltzmann $b(x)$; (d) Energy Maxwell-Boltzmann $k(x)$; (e) triangle $t(x)$.

Consider the uniform function (Figure 2a):

$$u(x) = 0.15 \tag{4}$$

Here we have an infinite area overall, no peak, and a very large tail. All links, regardless of the eigenvalues, have a probability of 0.15. This particular value is chosen to give it a similar area as the other basic functions within our main zone of interest. The other functions that extend to $+\inf$ all converge to 0. The zone we are mainly interested in here is $x \in [0, 3]$.

Consider the Gaussian distribution (Figure 2b), as used in Faulkner et al. (2016):

$$g_\sigma(x) = \frac{1}{\sqrt{2\sigma^2\pi}} \exp \frac{-x^2}{2\sigma^2} \tag{5}$$

Here we take $\sigma = 1$, so use $g = g_1$. Our prior use of the unit standard deviation Gaussian explains the choice of constants in the other functions here, to roughly match area and cut-off. This function has a peak at zero, so near equal eigenvalues have the highest probability of linking. It has a slightly larger area in our zone of interest, $x \in [0, 3]$, than does u. It has a long but quickly diminishing tail, so eigenvalue pairs

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

with large differences can link (unlike with f), but with a quickly decreasing probability.

Consider the Maxwell-Boltzmann velocity distribution: (Figure 2c):

$$b_a(x) = \sqrt{\frac{2}{\pi}} \frac{4x^2}{a^3} \exp \frac{-4x^2}{2a^2} \tag{6}$$

Here we use $b = b_{1.5}$. This has a similar area to g, but has a shifted peak. So linking is more likely with pairs of eigenvalues that are similar but not the same.

Consider the alternative Maxwell-Boltzmann distribution, $k = k_{0.23}$; this has a different tail shape due to the change to the exponential. This requires different variables to match the peak height and rough area of the other base functions. (Figure 2(d)):

$$k_a(x) = \sqrt{\frac{2}{\pi}} \frac{0.04x^2}{a^3} \exp \frac{-0.2x}{2a^2} \tag{7}$$

This has a fatter tail, to see the effect of a large tail and a shifted peak.

Finally, consider the triangle function, which we use to examine the separate effects of long tails and non-zero peaks (Figure 2e):

$$t(x) = \begin{cases} \frac{4}{15}x & : 0 < x \leq \frac{3}{2} \\ \frac{4}{15}(3-x) & : \frac{3}{2} < x \leq 3 \\ 0 & : 3 < x \end{cases} \tag{8}$$

This has a shifted peak at a similar position and height to b, but has no tail. We use it to assess the effects of long tails. It is also the only function in this set whose slope is discontinuous, which may allow us to assess the appropriateness of functions with discontinuous derivatives for our purposes.

Linking in Jordan Algebra AChem

We test these psfs g, b, u, k, and t in our JA-AChem (Faulkner et al., 2016), to investigate how changes in peak and tail affect the behaviour of our system. Three probabilities contribute to the overall probability of linking in JA-AChem: *Xcoll*, p_a and p_s (and p_l which is a combination of others).

The initial occurrence of a probability is when we select the list of reactants for our link. We randomly sample with replacement from our well-mixed tank, using a uniform distribution. After we sample the first two components, we have a probability for sampling further reactants, to produce an n-tad Jordan product link. We continue to sample, for one reactant at a time, with success probability *Xcoll*, until we fail. This gives a small but non-zero probability for attempting links with more than two reactants. Here we consider this to be part of the algorithm of how our tank functions, rather than an aspect of the rules of particle linking, and we set *Xcoll* = 0.2.

If we have more than two reactants in our link attempt, we take the reactants in sampling order, and consider the linking probabilities between each neighbouring pair of reactants. The *minimum* of these is taken as the linking probability of the overall link. So the probability of success depends on the probability of the weakest link forming. For four reactants, this is:

$$p_l(A,B,C,D) = \min\{p_l(A,B), p_l(B,C), p_l(C,D)\} \tag{9}$$

This linking probability p_l comprises two probability terms, combined here, and used separately elsewhere. We relate these to two different analogies with natural chemistry. We have the strength or potential strength, p_s, of the link, and the relative orientation, p_a, of the particles.

The orientation probability is defined as:

$$p_a(A,B) = 1 - \frac{(\mathbf{v}_a \cdot \mathbf{v}_b) + 1}{2} \tag{10}$$

where $\mathbf{v}_a \cdot \mathbf{v}_b$ is the scalar (dot) product of the two unit eigenvectors. When the vectors are parallel, $p_a = 0$; when they are anti-parallel, $p_a = 1$.

This orientation probability is used in two ways. It first defines which pair of eigenvalues are used to calculate link strength. We select the pair of eigenvalues e_a, e_b (where e_a is one of the three eigenvalues of matrix A) such that their corresponding eigenvectors $\mathbf{v}_a, \mathbf{v}_b$ maximise p_a. We can interpret this as the best aligned set of eigenvalues. The orientation probability also contributes to the probability of the link actually occurring.

Once we have used p_a to select e_a and e_b, we calculate $x = |e_a - e_b|$. This value of x is used to calculate $p_s(A,B) = p_s(x)$, the strength of the link that the particles would form using these eigenvalues. As well as its part in forming links, p_s is also the strength of the link once formed, and is the probability of the link *not* decomposing on a decomposition attempt.

We define the probability of the link occurring to be:

$$p_l(A,B) = \max\{p_a(A,B), p_s(A,B)\} \tag{11}$$

So linking happens if the link will be strong or if the particles are particularly well aligned.

Here we compare the effect on the system's behaviour when we use each of the above functions, u, g, b, k, t, as p_s. For each function, we run 5 rounds of linking and decomposition phases. We initialise the tank with 69 atomic particles. A linking phase performs a number of linking attempts equal to the number of particles in the tank at the beginning of the phase. For each linking attempt, we select a sample of particles from the tank. If the attempt is successful, and if the resulting particle is a novel particle, it is added to the tank. The reactants are not removed, allowing them to take part in further reactions. Hence the tank contains one of each kind of atom, and one of each kind of composite particle

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

Algorithm 1 Linking and Decomposition in Well Mixed Tank

```
Tank := {69 atomic particles}
for 1 to no_of_rounds−1 do
    LINKING PHASE
    DECOMPOSITION PHASE
LINKING PHASE
collect data from Tank

procedure LINKING PHASE
    for 1 to size(Tank) do
        reactants := 2 random particles from Tank
        while RANDOM() < Xcoll do
            reactants += a random particle from Tank
        attempt to link reactants
        if successful and product is new then
            add new particle to Tank

procedure DECOMPOSITION PHASE
    for all particles in Tank do
        attempt to decompose particle
        if successful and any product is new then
            add new particle(s) to Tank
```

made. This particular system does not have mass conservation, as here the "tank" is just the collection of possible particles found by the system so far. We have no interest here in the frequency with which any particle is formed; our focus is on finding novel particles so only these are added to the tank.

A decomposition phase performs a decomposition attempt on particle in the tank. If this leads to any new particles, they are added to the end of the tank and we attempt to decompose them in this phase. (We do not perform a decomposition phase after our final linking phase due to computational memory limitations.) See algorithm 1.

Speed and Area

The area under the psf directly relates to the general probability of any particular link occurring. The larger the area the more links will occur and the stronger they will be. We can see the area then as a sense of 'speed' in this system. If we do not change the shape of the function but simply decrease its area, then we should see roughly the same behaviour. It would be slowed down, as fewer links will form and decompose per generation. Due to this we try to keep the area of our functions reasonably similar with in the zone of interest (here taken to be $x \in [0, 3]$). We do not keep the areas exactly the same, preferring to have the peak height, shape and position more equal across functions, e.g. matching triangle peak height and span in t to peak height and span of zone of interest in b.

From the areas shown in Table 1 we can see that the functions' varying areas have similar numbers of particles produced in each system, allow for a sensible comparison of results. It is important to note that Table 1 gives the area in our "zone of interest", $x \in [0, 3]$. The overall area of u is much larger than that of the other functions as it is constant along the entire x axis while the other functions all tend to 0.

Function	Area	Particles
u	0.45	1331
g	0.49865	1796
b	0.499433	1938
k	0.715772	1852
t	0.6	1900

Table 1: The area under each of the probability spawning functions (integrated between 0 and 3), and the number of unique particles produced on running algorithm 1 including the 69 atoms.

Investigating Functions' Influence on System

In this section we look at the influence of these functions on different properties of the particles in our system. While we can collect data on many properties (such as total number of atoms in particles, number of distinct atoms in particles, ...), we focus here on three properties that show the most variation across the shape of the functions used. Other properties vary primarily with the size of the system. Systems that generate more particles tend to having bigger particles (ones with more atoms in them).

The three properties reported here are:

1. **Largest Link:** The largest number of particles involved in a single link within a particle (a link is formed by an n-tad Jordan product involving n particles).

2. **Strength:** The product of the strengths of each link in the composite particle:

$$P_s = \sum_{l \in P} l_s \tag{12}$$

   Atomic particles have no strength as they have no link, and are excluded from the strength statistics

3. **Self-synthesis:** The number of times in each particle there is a link linking two particles (atoms or composites) of the same kind

We consider how two features of our psfs may have affected each of these properties in our system:

(a) **Peak position:** In our base functions there are three different sorts of peak positions: none (u), zero (g) and non-zero (b,k and t).

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

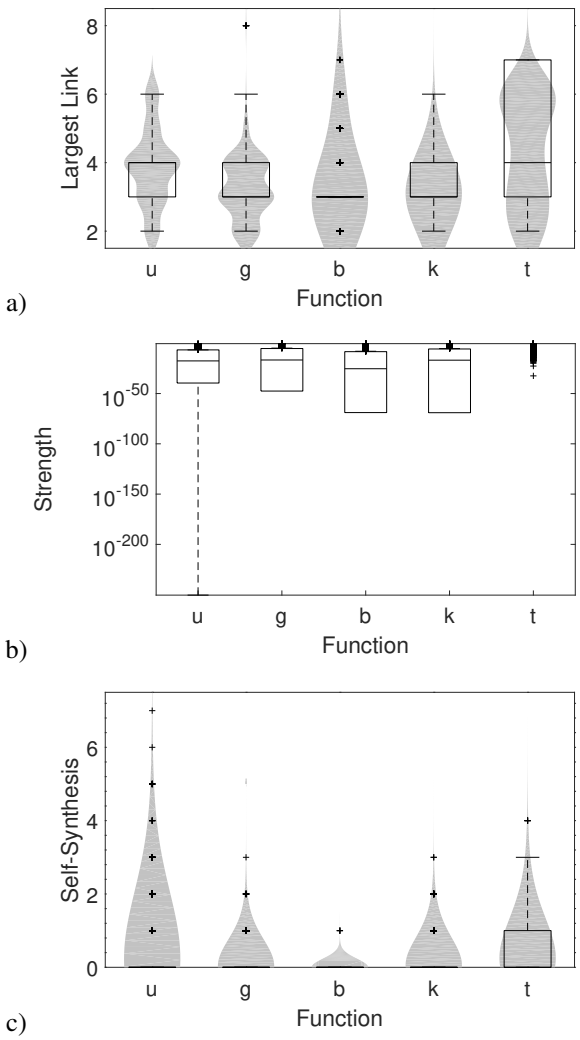

Figure 3: How particle properties vary with probability spawning function: (a) largest number of components in a link in each particle; (b) log-strength of the particles; (c) number of occurrences of self-synthesis in each particle.

(b) **Tail size/length:** We have four different sorts of tail in our functions: constant large (u), constant zero (t), small (g, b, $O(\exp -x^2)$) and large (k, $O(\exp -x)$).

In comparing these functions we are looking for how their features may be contributing to "interesting" distributions of particle properties. Here we are looking for wide distributions of values, preferably with outliers and a wide interquartile range. There should be a strong sense of median or common particle properties, but there should also be outliers and less common particles that stretch over a large range, indicating a rich variation in particles produced.

	g		b		t		k	
u	**0**	**0.64**	**0**	**0.64**	**0**	0.58	**0**	**0.67**
g	-	-	0.07	-	**0**	**0.67**	**0.003**	0.53
b	-	-	-	-	**0**	**0.65**	0.835	-
t	-	-	-	-	-	-	**0**	**0.69**

Table 2: p-values (left column) and effect sizes (right column) for size of largest link. Statistically significant p values ($p < 0.05$) are shown in bold; effect size A values are calculated for these; medium or larger effect sizes ($0.64 \leq A$) are shown in bold.

Largest Link

We start with how our functions affect the ability of our system to form larger (and possibly more complex) particles by looking at the largest links (Figure 3(a)).

Given a null hypothesis that psf has no effect on the number of particles in the largest link, we calculate the p-values, using the ranksum test, and the effect size, using the A test (Vargha and Delaney, 2000), between the tanks produced with each function (Table 2). These results split our functions into two groups whose distributions only have small differences: u and t; and g, b and k

For g, b and k there is very little spread and almost all particles have a maximum of 4 components in a link, but a few have as many as 7 or 8. Looking at u and t we see they have larger interquartile ranges, but spanning a different range of values: half of the particles produced with u have a largest link size of 3–4, and half of those produced with t have 4–6.

These groupings do match any particular peak position but do group to separate our constant and non-constant tails.

Strength

Since many of our particles have very small strength we use a log scale to make the distributions clearer (Figure 3b). All of u, g, b and k have skewed distribution, with the largest range being with k.

Notably t has a near non-existent distribution. Most of the particles in the t system have zero strength, and so a probability of 1 of decomposing. This is due to the constant zero tail. The selection of eigenvalues is based on alignment (see earlier), not on the similarity in value. This means links can occur with large differences in values, that is, large values of x. This results in low strength with long tails, but zero strength with no tail. So the tails on our distributions are important for being able to generate stable particles.

There is no indication that peak position has any influence on strength in the system.

Self-synthesis

There may be an advantage to a lack of tail. All the distributions have at least one instance of self-synthesis, but a

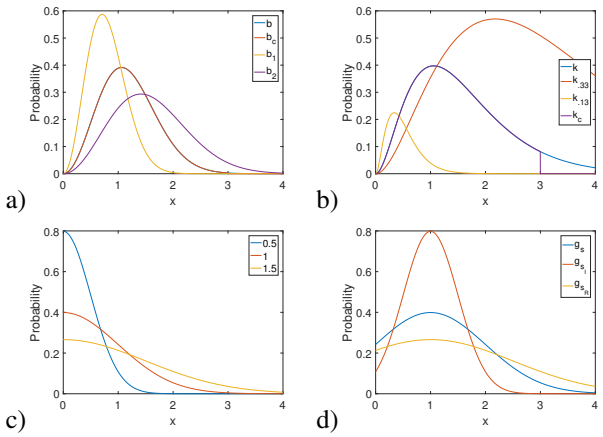

Figure 4: Sets of variations of base functions: (a) four variants of b; (b) four variants of k; (c) three variants of g; (d) four variants of g based of on the shifted peak variant: g_s.

median of zero (Figure 3c). The only function that produces significant self-synthesis is t.

The u, g and k functions also have at least one instance of a particle with more than one link with self-synthesis.

Again there is no connection to peak position, but there is a strong indication that having no tail affects this property.

Variations of Functions

In order to further investigate the above results we need a larger number of functions with different features. We now further test the effect of constant zero tails, and of tail size in general. We look at four sets of variations on our existing functions:

1. b: $b_{1.5}$ (original), and b_1, b_2, b_c, where

$$b_c = \begin{cases} b_{1.5} & x < 3 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

2. k: $k_{0.23}$ (original), and $k_{0.13}, k_{0.33}, k_c$, where

$$k_c = \begin{cases} k_{0.23} & x < 3 \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

3. g: g_1 (original), and $g_{0.5}, g_{1.5}$

4. shifted gaussian, gs: $gs_{0.5}$, gs_1, $gs_{1.5}$, where

$$gs_\sigma = \frac{1}{\sqrt{2\sigma^2\pi}} \exp \frac{(x-1)^2}{2\sigma^2} \tag{15}$$

The areas under these curves are not particularly similar, however any area effects caused by this difference should be identifiable across all four sets of variations so normalisation is not required. By allowing the areas to change we also allow ourselves to investigate if there are any area effects not previously noted.

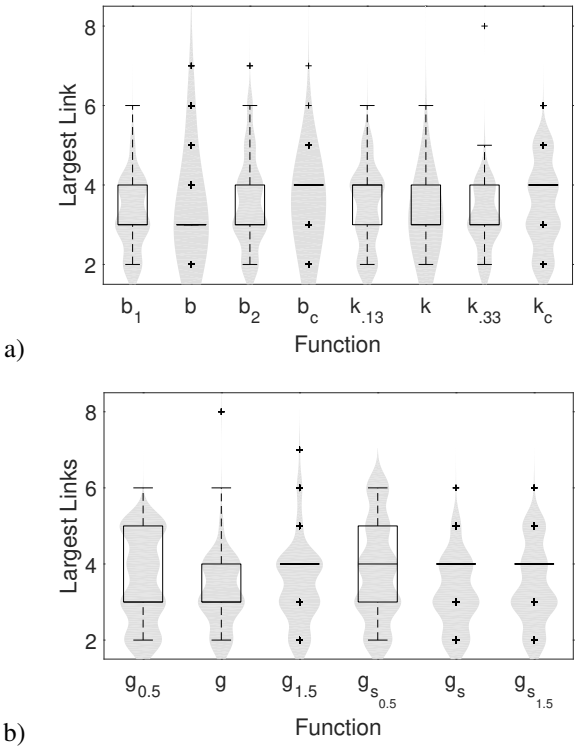

Figure 5: Distribution of largest link sizes with: (a) variants of b and k; (b) variants of g

Largest Link

Although the results in Figure 5 show statistically significant differences ($p < 0.05$), these all have small effect size ($A < 0.61$).

So the size of the largest link in each particle is not affected by the area under the curve, its steepness, the position and height of its peak or the size of its tail. This means there must be some other property involved that effects the size of the largest links. The largest effect size between any pair of variations is between gs and $gs_{0.5}$, with $A = 0.61$.

Strength

The results shown in Figure 6 show near non-existence of variation in the strength values of both b_c, with the cut-off, and b_1, the larger area b function. This agrees with the previous result: b_c has no tail, and, because of the increased steepness of the curve in b_1, it has much less of a tail than the other functions.

We can also see that b_2, with its larger tail, has a larger range of strengths than the base b functions. This further supports the idea that a longer tail results in a wider variety of link strengths.

When we look at the k variants we see k_c and $k_{.13}$, the smaller of the functions, both have low variation in strength. However $k_{.33}$ has a larger tail than k—much like b_2 and b—

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

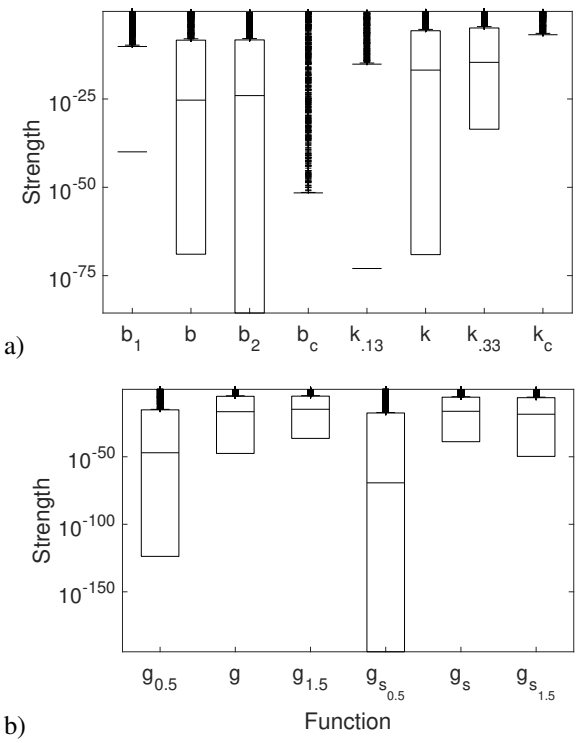

Figure 6: Strengths of particles for: (a) variants of b and k, (b) variants of g

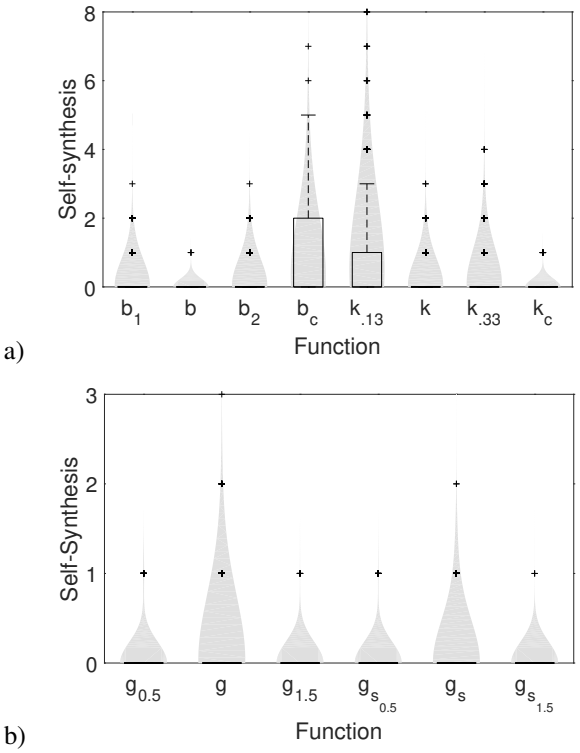

Figure 7: Occurrences of Self-synthesis with: (a) variants of b and k, (b) variants of g

but a smaller range of strengths, suggesting the spread of strengths is not simply correlated with tail size.

Our g variants further break our established pattern. $g_{0.5}$ has the smallest tail of the g variants, but the largest range of values. $g_{1.5}$ with the largest tail has the smallest range of strength values. This is contrary to our results look at just the base functions but agrees with our k results. The b and g functions have similar tails, so clearly something other than the tail has a strong effect on strength.

The large range found on the $g_{0.5}$ strengths might be connected to the larger range of values the function has.$g_{0.5}$ has a higher peak than g and a larger range of strengths. However b_1 also has a higher peak but has a very limited range of values.

The strength distributions of our particles seem to be more complicated than a simple aspect of tails or peak position, though in some cases there is still a correlation.

Self-synthesis

The self-synthesis results are shown in Figure 7. These show another effect of the lack of tail: an increased rate of self-synthesis. The only variant of b with any range of self-synthesis is b_c, which also has far more outliers than in the other functions.

The k variations seem to be anomolous until we consider the overall shape of $k_{.13}$. This function has a very small area and a very long tail meaning that because it has a very small tail that starts early it acts more like a function with no tail than our other functions. On the otherhand k_c has a very little self-synthesis. The reasons for this are not clear but it is the only function with a high value before the cut-off which may be influencing this.

Parameter changes have very little effect on self-synthesis in g functions, Figure 7b.

Summary

We have developed a set of probability spawning functions for use in linking in Jordan Algebra Artificial Chemistries. These have covered a wide range of options in terms of peak height, area, tail length, and height. These have produced various and different effects in our system. Some of these effects correlate with particular features, such as the tail cut off seeming to be connected to an increase in self-synthesis.

We have also found that there is a lot of complexity in our system. We see that different sets of functions can have very different effects. There are complex interactions between our psfs and the rest of our system.

Further Work

Here we have investigated only one of the probabilities used in our AChem: that used to give the link strength. There are

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).

other probabilities that should be similarly investigated, and potentially tuned, to give different behaviours. For example, parameters to the functions, such as a and σ, could be coupled to states of the system, such as temperature, to allow the probabilities to change with the system state.

The orientation probability p_a (eqn.10) is a candidate for variation. It has a role in both the overall linking probability p_l, and, crucially, in the choice of pair of eigenvalues. Currently we chose the eigenvalues and vectors that maximise p_a. We might instead chose by minimising p_a, or weight the choice in a less deterministic manner. Allowing the system, at times, to use less optimally "aligned" eigenvalue pairs may produce weaker or stronger bonds between the same particles. This would allow hard-to-form but then very strong links to occur. It would also provide weaker versions of particles. Because of their weaker links, these may be better at enabling catalysis by linking to further particles before decomposing.

Currently p_l uses the maximum of its two probabilities. Again, a different choice, such as the minimum, or some other combination of the two functions, would give different system behaviour.

We see there are many potential ways of combining probabilities. Further work will investigate how algebraically-defined combinations of psfs could be used to give combined properties that allow finer tuning of our system, for example, taking the maximum of the high-peaked $g_{0.5}$ and the large-tailed b_2 to give both properties.

Acknowledgements

PF is funded by a University of York, Department of Chemistry Teaching Studentship.

References

Clark, E. B., Hickinbotham, S. J., and Stepney, S. (2017). Semantic closure demonstrated by the evolution of a universal constructor architecture in an artificial chemistry. *J. R. Soc. Interface*, 14(130).

Faulconbridge, A., Stepney, S., Miller, J. F., and Caves, L. (2010). RBN-world: The hunt for a rich AChem. In *ALife XII, Odense, Denmark, August 2010*, pages 261–268. MIT Press.

Faulconbridge, A., Stepney, S., Miller, J. F., and Caves, L. S. D. (2011). RBN-World: A sub-symbolic artificial chemistry. In *ECAL 2009, Budapest, Hungary, September 2009*, volume 5777 of *LNCS*, pages 377–384. Springer.

Faulkner, P., Krastev, M., Sebald, A., and Stepney, S. (2017). Sub-symbolic artificial chemistries. In Stepney, S. and Adamatsky, A., editors, *Inspired by Nature*. Springer. (in press).

Faulkner, P., Sebald, A., and Stepney, S. (2016). Jordan algebra AChems: Exploiting mathematical richness for open ended design. In *Proceedings of the Artificial Life Conference 2016*, pages 582–589. MIT Press.

Hickinbotham, S., Clark, E., Stepney, S., Clarke, T., Nellis, A., Pay, M., and Young, P. (2012). Specification of the stringmol chemical programming language version 0.2. Technical Report YCS-2010-458, Department of Computer Science, University of York.

Hutton, T. J. (2002). Evolvable self-replicating molecules in an artificial chemistry. *Artificial Life*, 8(4):341–356.

Krastev, M., Sebald, A., and Stepney, S. (2016). Emergent bonding properties in the Spiky RBN AChem. In *ALife 2016, Cancun, Mexico, July 2016*, pages 600–607. MIT Press.

Krastev, M., Sebald, A., and Stepney, S. (2017). Functional grouping analysis of varying reactor types in the Spiky RBN AChem. In *ECAL 2017, Lyon, France, September 2017*. MIT Press.

McCrimmon, K. (2006). *A taste of Jordan algebras*. Springer Science & Business Media.

Ono, N. and Ikegami, T. (2001). Artificial chemistry: Computational studies on the emergence of self-reproducing units. In *6th European Conference on Artificial Life*, volume 2159 of *LNCS*, pages 186–195. Springer.

Sayama, H. (2009). Swarm chemistry. *Artificial Life*, 15(1):105–114.

Vargha, A. and Delaney, H. D. (2000). A critique and improvement of the "CL" common language effect size statistics of McGraw and wong. *J. Educ. Behav. Stat.*, 25(2):101–132.

Young, T. J. and Neshatian, K. (2015). The effect of reactant and product selection strategies on cycle evolution in an artificial chemistry. In *Artificial Life and Computational Intelligence*, volume 8955 of *LNCS*, pages 310–322. Springer.

Carole Knibbe et al, eds., Proceedings of the ECAL 2017, Lyon, France, 4-8 September 2017,
(Cambridge, MA: The MIT Press, ©2017 Massachusetts Institute of Technology).